COBOL & Z/OS Interview Questions

1. COBOL Programs

Q: What is the purpose of the WORKING-STORAGE SECTION in COBOL?

A: The WORKING-STORAGE SECTION is used to declare variables that retain their value throughout the program execution. These are static variables initialized when the program starts and hold their value unless explicitly changed.

Q: What is a copybook in COBOL?

A: A copybook is a reusable COBOL code component that typically defines record layouts or data structures. It is included in a COBOL program using the COPY statement.

Q: What is the difference between CALL BY REFERENCE and CALL BY CONTENT?

A: CALL BY REFERENCE passes the variable's address so the subprogram can modify the original value. CALL BY CONTENT passes a copy, so the original remains unchanged.

Q: What is a subprogram, and how do you call it?

A: A subprogram is a separate COBOL program invoked using CALL 'PROGNAME' USING var1, var2. Calls can be static or dynamic.

Q: What is the difference between WORKING-STORAGE and LINKAGE SECTION?

A: WORKING-STORAGE is for local variables; LINKAGE SECTION is used to access data passed from other programs or from JCL.

2. Z/OS Concepts & Utilities

Q: What is Z/OS, and how does it relate to COBOL?

A: Z/OS is IBM's mainframe OS designed for high-performance workloads. COBOL programs run on Z/OS using components like TSO/ISPF, JES, datasets, and utilities.

Q: Explain JES (Job Entry Subsystem) in Z/OS.

A: JES manages job submission, scheduling, and output processing. It handles JCL input, routes jobs, and stores output in spool datasets.

Q: What is the use of IEFBR14, IDCAMS, and IEBGENER?

A: IEFBR14: Dummy utility. IDCAMS: Manage VSAM/datasets. IEBGENER: Copy datasets, often used in file transfer.

Q: How can you merge two datasets using SORT?

A: Use SORT or MERGE with appropriate control fields to combine datasets. Example: SORT

FIELDS=(1,10,CH,A).

Q: How do you use JCL utilities to manage datasets?

A: Use IDCAMS for VSAM, IEFBR14 for allocation/deletion, and IEBGENER/SORT for copying or formatting data.

3. Restartable Programs

Q: What is a restartable COBOL program?

A: A program that can resume processing from a checkpoint after failure, without reprocessing completed records.

Q: How do you design a restartable program?

A: By implementing checkpoint-restart logic and storing the last processed key in a file or table.

Q: What is a checkpoint, and how is it used?

A: A saved state (e.g., record key) allowing the program to resume from that point after failure.

Q: How do you ensure no duplicate records are processed after restart?

A: Store the last key, skip already processed records, and compare keys before processing.

Q: Can you give a simple logic for restart?

A: READ INPUT-FILE AT END MOVE 'Y' TO EOF-SWITCH. IF RECORD-KEY > CHECKPOINT-KEY THEN PROCESS AND UPDATE.